

A Proof of the Pumping Lemma for Context-Free Languages Through Pushdown Automata

Antoine Amarilli* Marc Jeanmougin*

July 13, 2012

Abstract

The pumping lemma for context-free languages is a result about pushdown automata which is strikingly similar to the well-known pumping lemma for regular languages. However, though the lemma for regular languages is simply proved by using the pigeonhole principle on deterministic automata, the lemma for pushdown automata is proven through an equivalence with context-free languages and through the more powerful Ogden's lemma. We present here a proof of the pumping lemma for context-free languages which relies on pushdown automata instead of context-free grammars.

1 Setting

The pumping lemma for regular languages is the following well-known result:

Theorem 1. *Let L be a regular language over an alphabet Σ . There exists some integer $p \geq 1$ such that, for every $w \in L$ such that $|w| > p$, there exists a decomposition $w = xyz$ such that:*

1. $|xy| \leq p$
2. $|y| \geq 1$
3. $\forall n \geq 0, xy^n z \in L$

The pumping lemma for context-free languages [BHPS61], also known as the Bar-Hillel lemma, is the following similar result:

Theorem 2. *Let L be a context-free language over an alphabet Σ . There exists some integer $p \geq 1$ such that, for every $w \in L$ such that $|w| > p$, there exists a decomposition $w = uvxyz$ such that:*

1. $|vxy| \leq p$
2. $|vy| \geq 1$
3. $\forall n \geq 0, uv^n xy^n z \in L$

*École normale supérieure, Paris, France.

One would expect the classical proofs of these results to be similar. However, this is not the case. The pumping lemma for regular languages [HU79] is usually proved through the equivalence between regular languages and finite automata by picking a deterministic automaton A which recognizes the language L ; we can then use the fact that the accepting path of any word w longer than the number of states of A must pass by the same state twice (by the pigeonhole principle), yielding the points at which we can decompose w . The pumping lemma for context-free languages, however, is usually derived from Ogden's lemma [Ogd68] which is itself proved by examining context-free grammars and not pushdown automata (using the equivalence of these two formalisms).

It seems reasonable to hope that the pumping lemma for context-free languages can be proved directly from the properties of pushdown automata, with no reference to context-free grammars. In the next section, we propose such a proof. Though the underlying ideas that we introduce in this proof are apparently part of the folklore, we are not aware of existing published work on this topic.

2 Proof

Let L be a context-free language over an alphabet Σ . Let A be a pushdown automaton which recognizes L , with stack alphabet Γ . We denote by $|A|$ the number of states of A . To simplify the reasoning, we will impose the following condition on A : all transitions of A pop the topmost symbol of the stack and either push no symbol on the stack or push on the stack the previous topmost symbol and some other symbol. It is easy to see that any pushdown automata which pushes arbitrary sequences of symbols on the stack can be rewritten in this fashion by replacing its transitions by an initial pop transition followed by a sequence of ϵ -transitions pushing the appropriate symbols on the stack.

We define $p' = |A|^2|\Sigma|$ and define the pumping length to be $p = |A|(|\Gamma|+1)^{p'}$. We will now show that all $w \in L$ such that $|w| > p$ have a decomposition of the form $w = uvxyz$ such that $|vxy| \leq p$, $|vy| \geq 1$ and $\forall n \geq 0, uv^nxy^n z \in L$.

Let $w \in L$ such that $|w| > p$. Let π be an accepting path of minimal length for w (represented as a sequence of transitions of A), we denote its length by $|\pi|$. We can define, for $0 \leq i < |\pi|$, s_i the size of the stack at position i of the accepting path. For all $N > 0$, we will define an **N -level** over π as a set of three indices i, j, k with $0 \leq i < j < k \leq p$ such that the stack grows by N symbols between i and j and shrinks by N symbols between j and k . Formally, we require that:

1. $s_i = s_k, s_j = s_i + N$
2. for all n such that $i \leq n \leq j$, $s_i \leq s_n \leq s_j$
3. for all n such that $j \leq n \leq k$, $s_k \leq s_n \leq s_k$.

We define the level l of π as the maximal N such that π has an N -level. This definition is motivated by the following observation: if the size of the stack over a path π becomes larger than its level l , then the stack symbols more than l levels deep will never be popped. Formally, we define the **configurations** of A as the couples of a state of A and a sequence of l stack symbols (where stacks

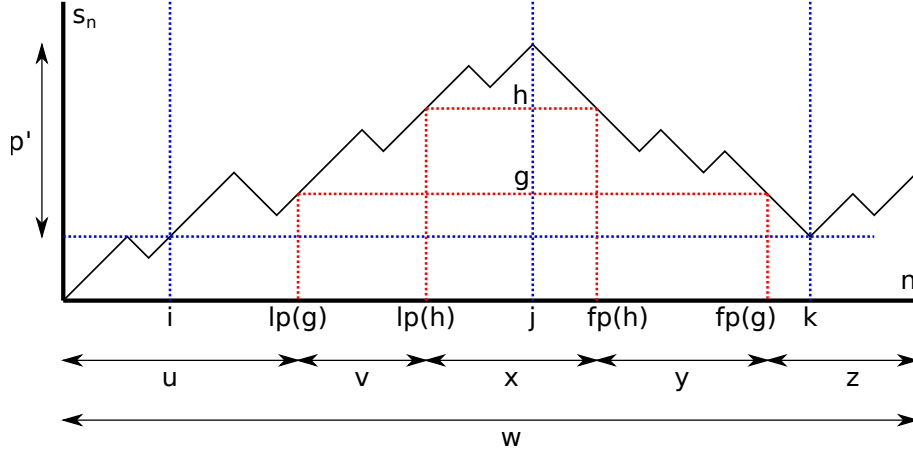


Figure 1: Illustration of the construction for case 2. To simplify the drawing, the distinction between the path positions and word positions are omitted.

of size less than l with be represented by padding them to l with a special blank symbol, which is why we use $|\Gamma| + 1$ when defining p). By definition, there are $|A|(|\Gamma| + 1)^l$ such configurations. Essentially, A acts as a finite automaton without stack between the configurations.

We can now distinguish two cases: either the level is low and the number of configurations is small, or the level is high. Formally:

1. $l < p'$ and, by the pigeonhole principle, the same configuration is encountered twice in the first $p + 1$ steps of π ,
2. $l \geq p'$ and, by the pigeonhole principle, we will prove that a certain notion of *full state* is repeated for two different stack sizes in any l -level of w .

Case 1. $l < p'$. In this case, the number of configurations is less than p . Hence, in the $p + 1$ first steps of π , the same configuration is encountered twice at two different positions, say $i < j$. Denote by \hat{i} (resp. \hat{j}) the position of the last letter of w read at step i (resp. j) of π . We have $\hat{i} \leq \hat{j}$. Hence, we can factor $w = uvxyz$ with $yz = \epsilon$, $u = w_{0 \dots \hat{i}}$, $v = w_{\hat{i} \dots \hat{j}}$, $x = w_{\hat{j} \dots |w|}$. (By $w_{x \dots y}$ we denote the letters of w from x inclusive to y exclusive.) By construction, $|vxy| \leq p$.

We also have to show that $\forall n \geq 0, uv^nxy^n z = uv^n x \in L$, but this follows from our observation above: stack symbols deeper than l are never popped, so there is no way to distinguish configurations which are equal according to our definition, and an accepting path for $uv^n x$ is built from that of w by repeating the steps between i and j , n times.

Finally, we also have $|v| > 0$, because if $v = \epsilon$, then, because we have the same configuration at steps i and j in π , $\pi' = \pi_{0 \dots i} \pi_{j \dots |\pi|}$ would be an accepting path for w , contradicting the minimality of π .

Case 2. $l \geq p'$. Let i, j, k be a p' -level. To any stack size h , $s_i \leq h \leq s_j$, we associate the **last push** $lp(h) = \max(\{y \leq j | s_y = h\})$ and the **first pop**

$\text{fp}(h) = \min(\{y \geq j \mid s_y = h\})$. By definition, $i \leq \text{lp}(h) \leq j$ and $j \leq \text{fp}(h) \leq k$. We say that the **full state** of a stack size h is the triple formed by:

1. the automaton state at position $\text{lp}(h)$
2. the topmost stack symbol at position $\text{lp}(h)$
3. the automaton state at position $\text{fp}(h)$

There are p' possible full states, and $p' + 1$ stack sizes between s_i and s_j , so, by the pigeonhole principle, there exist two stack sizes g, h with $s_i \leq g < h \leq s_j$ such that the full states at g and h are the same. Like in Case 1, we define by $\widehat{\text{lp}}(g)$, $\widehat{\text{lp}}(h)$, $\widehat{\text{fp}}(h)$ and $\widehat{\text{fp}}(g)$ the positions of the last letters of w read at the corresponding positions in π . We factor $w = uvxyz$ where $u = w_{0 \dots \widehat{\text{lp}}(g)}$, $v = w_{\widehat{\text{lp}}(g) \dots \widehat{\text{lp}}(h)}$, $x = w_{\widehat{\text{lp}}(h) \dots \widehat{\text{fp}}(h)}$, $y = w_{\widehat{\text{fp}}(h) \dots \widehat{\text{fp}}(g)}$, and $z = w_{\widehat{\text{fp}}(g) \dots |w|}$.

This factorization ensures that $|vxy| \leq p$ (because $k \leq p$ by our definition of levels).

We also have to show that $\forall n \geq 0, uv^nxy^n z \in L$. To do so, observe that each time that we repeat v , we start from the same state and the same stack top and we do not pop below our current position in the stack (otherwise we would have to push again at the current position, violating the maximality of $\text{lp}(g)$), so we can follow the same path in A and push the same symbol sequence on the stack. By the maximality of $\text{lp}(h)$ and the minimality of $\text{fp}(h)$, while reading x , we do not pop below our current position in the stack, so the path followed in the automaton is the same regardless of the number of times we repeated v . Now, if we repeat w as many times as we repeat v , since we start from the same state, since we have pushed the same symbol sequence on the stack with our repeats of v , and since we do not pop more than what v has stacked by minimality of $\text{fp}(g)$, we can follow the same path in A and pop the same symbol sequence from the stack. Hence, an accepting path from $uv^nxy^n z$ can be constructed from the accepting path for w .

Finally, we also have $|vy| > 1$, because like in case 1, if $v = \epsilon$ and $y = \epsilon$, we can build a shorter accepting path for w by removing $\pi_{\widehat{\text{lp}}(g) \dots \widehat{\text{lp}}(h)}$ and $\pi_{\widehat{\text{fp}}(h) \dots \widehat{\text{fp}}(g)}$.

Hence, we have an adequate factorization in both cases, and the result is proved.

References

- [BHPS61] Yehoshua Bar-Hillel, Micha A. Perles, and Eli Shamir. On formal properties of simple phrase structure grammars. *Zeitschrift für Phonetik, Sprachwissenschaft und Kommunikationsforschung*, (14):143–172, 1961.
- [HU79] John E. Hopcroft and Jefferey D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Adison-Wesley Publishing Company, Reading, Massachusetts, USA, 1979.
- [Ogd68] William F. Ogden. A helpful result for proving inherent ambiguity. *Mathematical Systems Theory*, 2(3):191–194, 1968.